

Отчёт по лабораторной работе №6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Щербак Маргарита Романовна

2022

1 Цель работы:

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

2 Теоретическое введение:

В системе по умолчанию открыто три специальных потока: – `stdin` — стандартный поток ввода (по умолчанию: клавиатура), файловый дескриптор 0; – `stdout` — стандартный поток вывода (по умолчанию: консоль), файловый дескриптор 1; – `stderr` — стандартный поток вывод сообщений об ошибках (по умолчанию: консоль), файловый дескриптор 2. Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода `stdout`. Например, команда `ls` выводит в стандартный поток вывода (консоль) список файлов в текущей директории. Потоки вывода и ввода можно перенаправлять на другие файлы или устройства. Проще всего это делается с помощью символов `>`, `>>`, `<`, `<<`.

Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

Чаще всего скрипты на `Bash` используются в качестве автоматизации каких-то рутинных операций в консоли, отсюда иногда возникает необходимость в обработке `stdout` одной команды и передача на `stdin` другой команде, при этом результат выполнения команды должен обработан.

3 Выполнение лабораторной работы:

1. Осуществили вход в систему, используя соответствующее имя пользователя(Рис. 3.1)

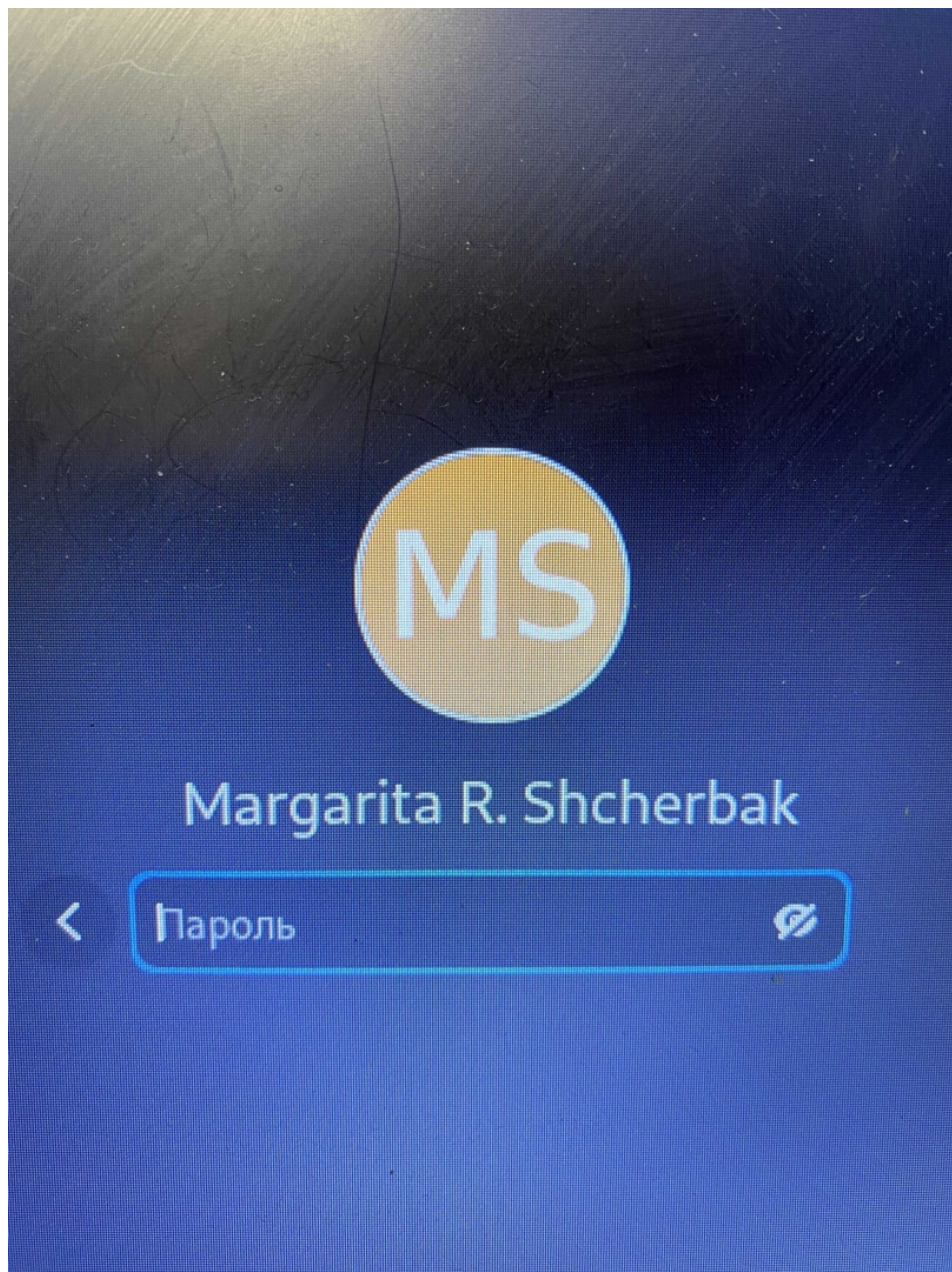


Рис. 3.1: Вход в систему

2. Записали в файл file.txt названия файлов, содержащихся в каталоге /etc. Дописали в этот же файл названия файлов, содержащихся в моём домашнем каталоге. (Рис. 3.2 - Рис. 3.3).

```

[mrshcherbak@fedora ~]$ ls
abcl  bin  may  my_os  reports  vvvvvvvvv  Видео  Загрузки  Музыка  'Рабочий стол'
australia  feathers  monthly  play  ski.plases  work  Документы  Изображения  Общедоступные  Шаблоны
[mrshcherbak@fedora ~]$ ls /etc > file.txt
[mrshcherbak@fedora ~]$ ls
abcl  bin  file.txt  monthly  play  ski.plases  work  Документы  Изображения  Общедоступные  Шаблоны
australia  feathers  may  my_os  reports  vvvvvvvvv  Видео  Загрузки  Музыка  'Рабочий стол'

```

Рис. 3.2: Выполнение

```

[mrshcherbak@fedora ~]$ ls >>file.txt
[mrshcherbak@fedora ~]$ cat file.txt
abrt
adjtime
aliases
alsa
alternatives
anaconda
anthy-unicode.conf
appstream.conf
asound.conf
audit
authselect
avahi
bash_completion.d
bashrc
bindresvport.blacklist
binfmt.d
bluetooth
brlapi.key
brltty
brltty.conf
ceph
chkconfig.d
chromium

```

Рис. 3.3: Выполнение

3. Вывели имена всех файлов из file.txt, имеющих расширение .conf, после чего записали их в новый текстовый файл conf.txt (Рис. 3.4 - Рис. 3.5).

```
[mrshcherbak@fedora ~]$ grep '\.conf$' file.txt
anthy-unicode.conf
appstream.conf
asound.conf
brltty.conf
chrony.conf
dleyna-renderer-service.conf
dleyna-server-service.conf
dnsmasq.conf
dracut.conf
extlinux.conf
fprintd.conf
fuse.conf
host.conf
idmapd.conf
jwhois.conf
kdump.conf
krb5.conf
ld.so.conf
libaudit.conf
libuser.conf
locale.conf
logrotate.conf
man_db.conf
mke2fs.conf
mtools.conf
nfs.conf
nfsmount.conf
nsswitch.conf
```

Рис. 3.4: Вывели имена всех файлов из file.txt, имеющих расширение .conf

```

sestatus.conf
sudo.conf
swtpm-localca.conf
swtpm_setup.conf
sysctl.conf
tcsd.conf
Trolltech.conf
updatedb.conf
uresourced.conf
usb_modeswitch.conf
vconsole.conf
xattr.conf
[mrshcherbak@fedora ~]$ grep '\.conf$' file.txt > conf.txt
[mrshcherbak@fedora ~]$ cat conf.txt
anthy-unicode.conf
appstream.conf
asound.conf
brltty.conf
chrony.conf
dleyna-renderer-service.conf
dleyna-server-service.conf
dnsmasq.conf
dracut.conf
extlinux.conf
fprintd.conf
fuse.conf
host.conf
idmapd.conf
jwhois.conf
kdump.conf

```

Рис. 3.5: записали имена всех файлов из file.txt, имеющих расширение .conf в новый текстовый файл conf.txt

4. Определили, какие файлы в моём домашнем каталоге имеют имена, начинавшиеся с символа с. (Рис. 3.6)

```

[mrshcherbak@fedora ~]$ ls | grep ^c
conf.txt
[mrshcherbak@fedora ~]$ ls c*
conf.txt

```

Рис. 3.6: Файлы, имеющие имена, начинавшиеся с символа с

5. Вывели на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h. (Рис. 3.7 - Рис. 3.8).


```
[mrshcherbak@fedora ~]$ find /etc -maxdepth 1 -name "h*" | less
```

Рис. 3.7: Выполнение

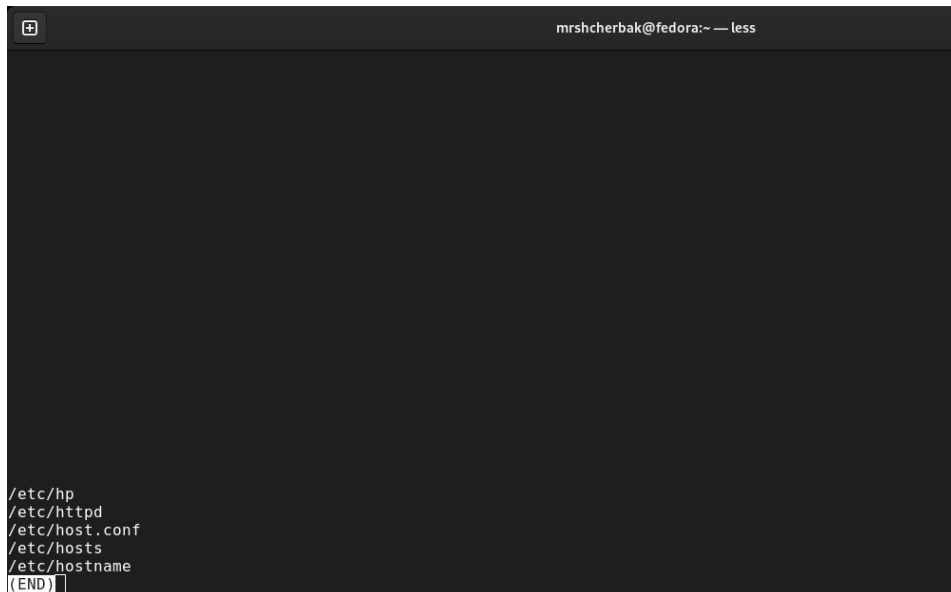


Рис. 3.8: Имена файлов из каталога /etc, начинающиеся с символа h.

6. Запустили в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.
7. Удалили файл ~/logfile. (Рис. 3.9).

```
[mrshcherbak@fedora ~]$ find ~ -name "log*" -print >logfile &  
[1] 39763  
[mrshcherbak@fedora ~]$ rm logfile  
[1]+  Завершён      find ~ -name "log*" -print > logfile  
[mrshcherbak@fedora ~]$
```

Рис. 3.9: Выполнение

8. Запустили из консоли в фоновом режиме редактор gedit.
9. Определили идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. (Рис. 3.10 - Рис. 3.12).

```
[mrshcherbak@fedora ~]$ ps aux | grep gedit | grep -v grep  
mrshche+ 39827  1.1  2.0 855836 73052 pts/0    Sl   14:37   0:01 gedit  
[mrshcherbak@fedora ~]$
```

Рис. 3.10: Использование команды ps, конвейера и фильтра

```
[mrshcherbak@fedora ~]$ ps | grep -i gedit
39827 pts/0    00:00:01 gedit
[mrshcherbak@fedora ~]$ ps
  PID TTY          TIME CMD
 3984 pts/0    00:00:00 bash
 3987 pts/0    00:00:01 gedit
 3989 pts/0    00:00:00 ps
```

Рис. 3.11: Определение идентификатора процесса

```
[mrshcherbak@fedora ~]$ pgrep -f gedit
39827
[mrshcherbak@fedora ~]$
```

Рис. 3.12: Идентификатор процесса gedit

10. Прочитали справку (man) команды kill, после чего использовали её для завершения процесса gedit. (Рис. 3.13).

```
[mrshcherbak@fedora ~]$ man kill
[mrshcherbak@fedora ~]$ kill 39827
[1]+  Завершено      gedit
[mrshcherbak@fedora ~]$
```

Рис. 3.13: Завершение процесса gedit

11. Выполнили команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man. (Рис. 3.14 - Рис. 3.17).

```
mrshcherbak@fedora:~ — man df
DF(1)                                User Commands                                DF(1)

NAME
df - report file system disk space usage

SYNOPSIS
df [OPTION]... [FILE]...

DESCRIPTION
This manual page documents the GNU version of df. df displays the amount of disk space available on the file system containing each file name argument. If no file name is given, the space available on all currently mounted file systems is shown. Disk space is shown in 1K blocks by default, unless the environment variable POSIXLY_CORRECT is set, in which case 512-byte blocks are used.

If an argument is the absolute file name of a disk device node containing a mounted file system, df shows the space available on that file system rather than on the file system containing the device node. This version of df cannot show the space available on unmounted file systems, because on most kinds of systems doing so requires very nonportable intimate knowledge of file system structures.

OPTIONS
Show information about the file system on which each FILE resides, or all file systems by default.

Mandatory arguments to long options are mandatory for short options too.

-a, --all
    include pseudo, duplicate, inaccessible file systems

-B, --block-size=SIZE
    scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of 1,048,576 bytes; see SIZE format below

Manual page df(1) line 1 (press h for help or q to quit)
```

Рис. 3.14: man df

```
[mrshcherbak@fedora ~]$ df
Файловая система 1K-блоков  Исползовано  Доступно  Исползовано%  Смонтировано в
devtmpfs          1732876            0   1732876            0% /dev
tmpfs             1753260            0   1753260            0% /dev/shm
tmpfs             701308            1428   699880            1% /run
/dev/sda2         46710784   13969460   31872124        31% /
tmpfs             1753264            436   1752828            1% /tmp
/dev/sda2         46710784   13969460   31872124        31% /home
/dev/sda1         996780            175300   752668           19% /boot
tmpfs             350652            1044   349608            1% /run/user/1000
/dev/sr0          60096            60096            0       100% /run/media/mrshcherbak/VBox_GAs_6.1.34
Labss            248890364   198757176   50133188         80% /media/sf_Labss
[mrshcherbak@fedora ~]$
```

Рис. 3.15: Выполнение команды df

```
mrshcherbak@fedora:~ — man du
DU(1)                                User Commands                                DU(1)

NAME
du - estimate file space usage

SYNOPSIS
du [OPTION]... [FILE]...
du [OPTION]... --files0-from=F

DESCRIPTION
Summarize disk usage of the set of FILES, recursively for directories.

Mandatory arguments to long options are mandatory for short options too.

-0, --null
    end each output line with NUL, not newline

-a, --all
    write counts for all files, not just directories

--apparent-size
    print apparent sizes, rather than disk usage; although the apparent size is usually smaller, it may be larger due to holes in ('sparse') files, internal fragmentation, indirect blocks, and the like

-B, --block-size=SIZE
    scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of 1,048,576 bytes; see SIZE format below

-b, --bytes
    equivalent to '--apparent-size --block-size=1'

Manual page du(1) line 1 (press h for help or q to quit)
```

Рис. 3.16: man du

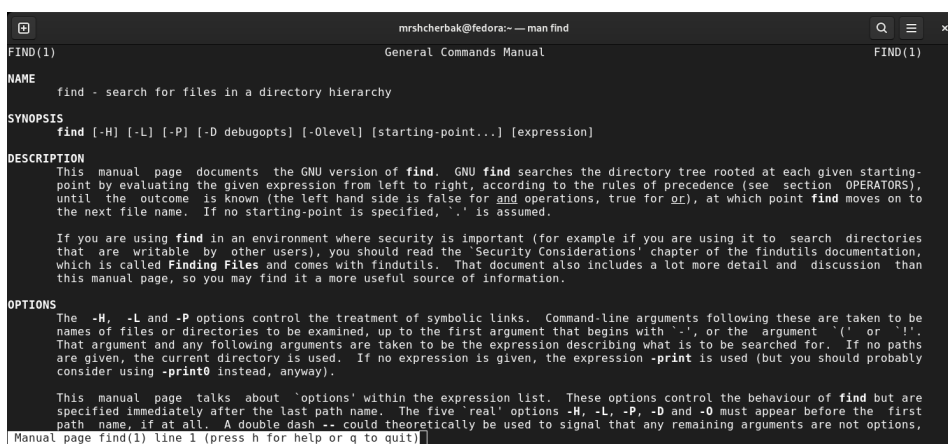
```

[mrshcherbak@fedora ~]$ man du
[mrshcherbak@fedora ~]$ du bin
54940      bin
[mrshcherbak@fedora ~]$ du -h bin
54M       bin
[mrshcherbak@fedora ~]$ 

```

Рис. 3.17: Выполнение команды du

12. Воспользовавшись справкой команды find, вывели имена всех директорий, имеющих в своём домашнем каталоге. (Рис. 3.18 - Рис. 3.19).



```

FIND(1)                                General Commands Manual                                FIND(1)

NAME
    find - search for files in a directory hierarchy

SYNOPSIS
    find [-H] [-L] [-P] [-D debugopts] [-O level] [starting-point...] [expression]

DESCRIPTION
    This manual page documents the GNU version of find. GNU find searches the directory tree rooted at each given starting-point by evaluating the given expression from left to right, according to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand side is false for and operations, true for or), at which point find moves on to the next file name. If no starting-point is specified, '.' is assumed.

    If you are using find in an environment where security is important (for example if you are using it to search directories that are writable by other users), you should read the 'Security Considerations' chapter of the findutils documentation, which is called Finding Files and comes with findutils. That document also includes a lot more detail and discussion than this manual page, so you may find it a more useful source of information.

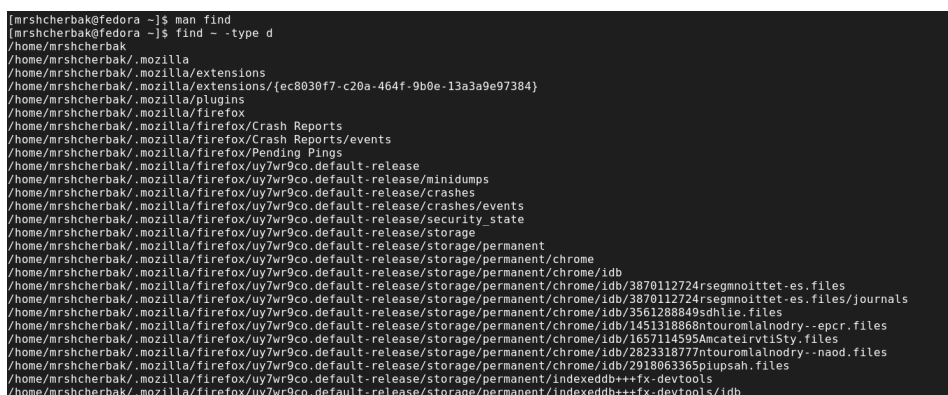
OPTIONS
    The -H, -L and -P options control the treatment of symbolic links. Command-line arguments following these are taken to be names of files or directories to be examined, up to the first argument that begins with '.', or the argument '(' or '!'. That argument and any following arguments are taken to be the expression describing what is to be searched for. If no paths are given, the current directory is used. If no expression is given, the expression -print is used (but you should probably consider using -print0 instead, anyway).

    This manual page talks about 'options' within the expression list. These options control the behaviour of find but are specified immediately after the last path name. The five 'real' options -H, -L, -P, -D and -O must appear before the first path name, if at all. A double dash -- could theoretically be used to signal that any remaining arguments are not options, consider using --help instead.

Manual page find(1) line 1 (press h for help or q to quit)

```

Рис. 3.18: Справка по команде find



```

[mrshcherbak@fedora ~]$ man find
[mrshcherbak@fedora ~]$ find -type d
/home/mrshcherbak
/home/mrshcherbak/.mozilla
/home/mrshcherbak/.mozilla/extensions
/home/mrshcherbak/.mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
/home/mrshcherbak/.mozilla/plugins
/home/mrshcherbak/.mozilla/firefox
/home/mrshcherbak/.mozilla/firefox/Crash Reports
/home/mrshcherbak/.mozilla/firefox/Crash Reports/events
/home/mrshcherbak/.mozilla/firefox/Pending Pings
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/minidumps
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/crashes
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/crashes/events
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/security_state
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files/journals
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/3561288849sdhlie.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/1451318868ntouromlalnodyr--epcr.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/1657114595AmcateirvtiSty.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/2823318777ntouromlalnodyr--naod.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/chrome/idb/2918063365piupsah.files
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/indexeddb+++fx-devtools
/home/mrshcherbak/.mozilla/firefox/uy7wr9co.default-release/storage/permanent/indexeddb+++fx-devtools/idb

```

Рис. 3.19: Выполнение

4 Контрольные вопросы:

1. Какие потоки ввода вывода вы знаете? Stdin и stdout, stderr Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода stdout.
2. Объясните разницу между операцией > и ».
'>' Перенаправление вывода в файл
'»' Перенаправление вывода в файл и открытие файла в режиме добавления (данные добавляются в конец файла).
3. Что такое конвейер? Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий:
команда1|команда2 (это означает, что вывод команды 1 передастся на ввод команде 2).
4. Что такое процесс? Чем это понятие отличается от программы? Процесс рассматривается операционной системой как заявка на потребление всех видов ресурсов, кроме одного – процессорного времени. Этот последний важнейший ресурс распределяется операционной системой между другими единицами работы – потоками, которые и получили свое название благодаря тому, что они представляют собой последовательности (потоки выполнения) команд. Процесс – это выполнение программы. Он считается активной сущностью и реализует действия, указанные в программе. Программа представляет собой статический набор команд, а процесс это набор ресурсов и данных, использующихся при выполнении программы.

5. Что такое PID и GID? `pid`: идентификатор процесса (PID) процесса (`processID`), к которому вызывают метод `gid`: идентификатор группы UNIX, в котором работает программа.
6. Что такое задачи и какая команда позволяет ими управлять? Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать знак амперсанда `&`. Например: `gedit &`.
7. Найдите информацию об утилитах `top` и `htop`. Каковы их функции? `Top` – отобразить запущенные процессы, используемые ими ресурсы и другую полезную информацию (с автоматическим обновлением данных) `htop` – это продвинутый консольный мониторинг процессов. Утилита выводит постоянно меняющийся список системных процессов, который сортируется в зависимости от нагрузки на ЦПУ. Если делать сравнение `ctop`, то `htop` показывает абсолютно все процессы в системе, время их непрерывного использования, загрузку процессоров и расход оперативной памяти.
8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды. `find` – это команда для поиска файлов и каталогов на основе специальных условий. Ее можно использовать в различных обстоятельствах, например, для поиска файлов по разрешениям, владельцам, группам, типу, размеру и другим подобным критериям. Команда `find` имеет такой синтаксис: `find[папка][параметры] критерий шаблон [действие]` Папка – каталог в котором будем искать Параметры – дополнительные параметры, например, глубина поиска, и т.д. Критерий – по какому критерию будем искать: имя, дата создания, права, владелец и т.д. Шаблон – непосредственно значение по которому будем отбирать файлы. Основные параметры:
 - `R` никогда не открывать символические ссылки
 - `L` - получает информацию о файлах по символическим ссылкам. Важно для дальнейшей обработки, чтобы обрабатывалась не ссылка, а сам файл.

- `maxdepth` - максимальная глубина поиска по подкаталогам, для поиска только в текущем каталоге установите 1.
 - `depth` - искать сначала в текущем каталоге, а потом в подкаталогах
 - `mount` искать файлы только в этой файловой системе.
 - `version` - показать версию утилиты `find`
 - `print` - выводить полные имена файлов
 - `typef` - искать только файлы
 - `typed` - поиск папки в Linux
 - `name` - поиск файлов по имени
 - `perm` - поиск файлов в Linux по режиму доступа
 - `user` - поиск файлов по владельцу `-group` - поиск по группе
 - `mtime` - поиск по времени модификации файла
 - `atime` - поиск файлов по дате последнего чтения
 - `nogroup` - поиск файлов, не принадлежащих ни одной группе
 - `nouser` - поиск файлов без владельцев
 - `newer` - найти файлы новее чем указанный
 - `size` - поиск файлов в Linux по их размеру
- Примеры:
- `find ~ -type d` поиск директорий в домашнем каталоге
- `find ~ -type f -name ".*"` поиск скрытых файлов в домашнем каталоге.

9. Можно ли по контексту (содержанию) найти файл? Если да, то как? Да, через команду `grep`. Например: `grep Aug -R /var/log/*` вывода строки, содержащие "Aug", во всех файлах, находящихся в директории `/var/log` и ниже.
«`grep -r`» слово/выражение, которое нужно найти».
10. Как определить объем свободной памяти на жёстком диске? Для определения объёма свободного пространства на файловой системе можно воспользоваться командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования.
11. Как определить объем вашего домашнего каталога? Команда `du` показывает

число килобайт, используемое каждым файлом или каталогом.

12. Как удалить зависший процесс Можно удалить через kill, написав айди процесса. Основные сигналы (каждый сигнал имеет свой номер), которые используются для завершения процесса:

- SIGINT–самый безобидный сигнал завершения, означает Interrupt. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш Ctrl+C. Процесс правильно завершает все свои действия и возвращает управление;
- SIGQUIT–это еще один сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей что нужно завершиться и программа может выполнить корректное завершение или проигнорировать сигнал. Сочетание клавиш Ctrl+Q;
- SIGHUP–сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом;
- SIGTERM–немедленно завершает процесс, но обрабатывается программой, поэтому позволяет ей завершить дочерние процессы и освободить все ресурсы;
- SIGKILL–тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными.

5 Выводы:

Таким образом, в ходе ЛРН№6 я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практически навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.